This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images,
Please do not report the images to the
Image Problem Mailbox.

THIS PAGE BLANK (USPTO)

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/60

A1

(11) International Publication Number: WO 97/32265

(43) International Publication Date: 4 September 1997 (04.09.97)

(21) International Application Number: PCT/US97/02718

(22) International Filing Date: 25 February 1997 (25.02.97)

(30) Priority Data:
08/609,694 1 March 1996 (01.03.96) US

(71) Applicant: ELECTRONIC DATA SYSTEMS CORPORATION [US/US]; 5400 Legacy Drive, M/S H3-3A-05, Plano, TX 75024 (US).

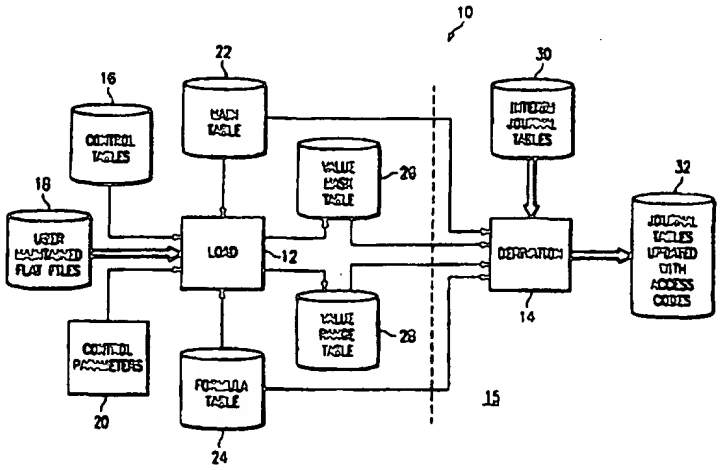
(72) Inventors: ROSE, Sandra, T.; 1703 Cross Point, McKinney, TX 75070 (US). ELSTROD, John, P.; 2804 Haun Trail, Plano, TX 75075 (US). ANDREWS, Patricia, C.; 3813 18th Street, Plano, TX 75074 (US). FORBES, Charles, G.; 1105 Coffeyville Court, Plano, TX 75023 (US). DONOVAN, Joseph, M.; 2333 Custa Lane, McKinney, TX 75070 (US). KLINGER, Lynn, M.; 7508 England Drive, Plano, TX 75025 (US).

(74) Agent: GRIEBENOW, L., Joy; Electronic Data Systems Corporation, 5400 Legacy Drive, M/S H3-3A-05, Plano, TX 75024 (US).

(81) Designated States: AU, CA, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: SYSTEM AND METHOD FOR DERIVING FINANCIAL RESPONSIBILITY IDENTIFICATION



(57) Abstract

A system and method (10) for deriving access codes for billing customers includes a load process (12) receiving a set of system attributes (20) and using at least one of the system attributes (20) for looking up in at least one lookup table (16, 18, 22, 24) for an access code and at least one formula value, and further storing the access code and at least one formula value in an access code lookup table (26, 28). The system (10) further includes a derivation process (14) which receives a specification of values for the set of system attributes from a billing record (30), uses the specification of values for looking up in the at least one lookup table (16, 18, 22, 24) for a derivation formula, uses the derivation formula for looking up in the billing record (30) for an attribute value, and further uses the attribute value for comparing with the formula value stored in the access code lookup table (26, 28). The access code associated with the formula value comparable to the attribute value is the derived access code.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AM	Armenia	GB	United Kingdom	MW	Malawi
AT	Austria	GE	Georgia	MX	Mexico
AU	Australia	GN	Guinea	NE	Niger
BB	Barbados	GR	Greece	NL	Netherlands
BE	Belgium	HU	Hungary	NO	Norway
BF	Burkina Faso	IE	Ireland	NZ	New Zealand
BG	Bulgaria	IT	Italy	PL	Poland
BJ	Benin	JP	Japan	PT	Portugal
BR	Brazil	KE	Kenya	RO	Romania
BY	Belarus	KG	Kyrgystan	RU	Russian Federation
CA	Canada	KP	Democratic People's Republic of Korea	SD	Sudan
CF	Central African Republic			SE	Sweden
CG	Congo	KR	Republic of Korea	SG	Singapore
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LR	Liberia	SZ	Swaziland
CS	Czechoslovakia	LT	Lithuania	TD	Chad
CZ	Czech Republic	LU	Luxembourg	TG	Togo
DE	Germany	LV	Latvia	TJ	Tajikistan
DK	Denmark	MC	Monaco	TT	Trinidad and Tobago
EE	Estonia	MD	Republic of Moldova	UA	Ukraine
ES	Spain	MG	Madagascar	UG	Uganda
FI	Finland	ML	Mali	US	United States of America
FR	France	MN	Mongolia	UZ	Uzbekistan
GA	Gabon	MR	Mauritania	VN	Viet Nam

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 97/02718

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5202989 A	13-04-93	JP 3025556 A	04-02-91

INTERNATIONAL SEARCH REPORT

Int ional Application No
PCT/US 97/02718

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 37, no. 1, January 1994, NEW YORK, US, pages 209-212, XP000644924 ANONYMOUS: "Method for Gathering Accounting Data on Active Resources" see the whole document	1,2,16

INTERNATIONAL SEARCH REPORT

Int ional Application No
PCT/US 97/02718

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 202 989 A (HIROSAWA TOSHIO ET AL) 13 April 1993 see abstract; claims 1,2 ---	1,2,6
A	IBM TECHNICAL DISCLOSURE BULLETIN, vol. 17, no. 4, September 1974, NEW YORK, US, pages 1029-1030, XP000616887 ANONYMOUS: "Comprehensive Computer Accounting System. September 1974." see the whole document --- -/--	1,2,16

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 June 1997	26. 06. 97

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Suendermann, R

Form PCT/ISA/210 (second sheet) (July 1992)

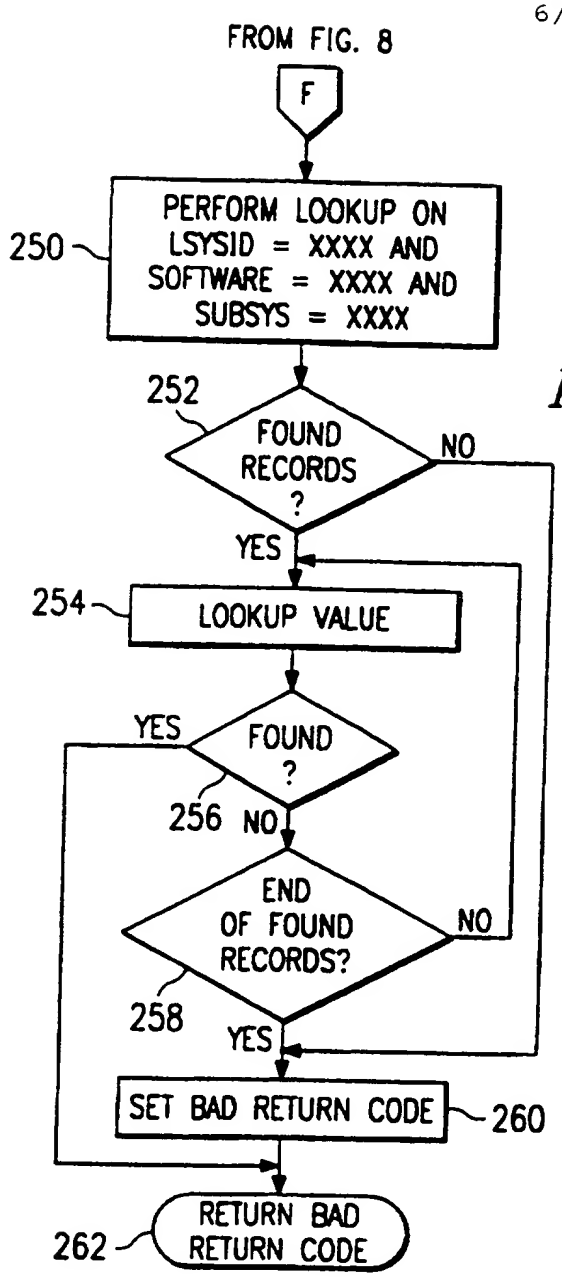

FROM FIG. 8
F
PERFORM LOOKUP ON LSYSID = XXXX AND SOFTWARE = XXXX AND SUBSYS = XXXX
250
252
FOUND RECORDS ?
NO
YES
254
LOOKUP VALUE
YES
FOUND ?
256
NO
END OF FOUND RECORDS?
NO
258
YES
SET BAD RETURN CODE
260
RETURN BAD RETURN CODE
262

FIG. 9

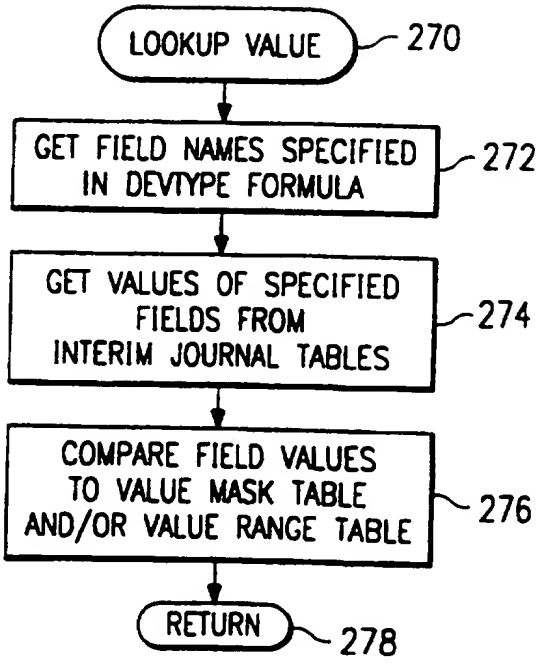

LOOKUP VALUE
270
GET FIELD NAMES SPECIFIED IN DEVTYPE FORMULA
272
GET VALUES OF SPECIFIED FIELDS FROM INTERIM JOURNAL TABLES
274
COMPARE FIELD VALUES TO VALUE MASK TABLE AND/OR VALUE RANGE TABLE
276
RETURN
278

FIG. 10

SYSTEM AND METHOD FOR DERIVING FINANCIAL RESPONSIBILITY IDENTIFICATION

TECHNICAL FIELD OF THE INVENTION

This invention is related in general to the field of information processing. More particularly, the invention is related to a system and method for deriving financial responsibility identification.

BACKGROUND OF THE INVENTION

For companies that provide information handling services in the way of computing and data processing to customers, means must be provided to access the data processing and computing costs in order to accurately allocate those costs to the customer accounts. This practice is known as "internal charge-back." In many instances, information handling services are provided by operating vast distributed programs on more than one computer, which are often located remotely from one another. Moreover, some companies that operate globally, the business units and their associated computer systems may be located in different parts of the world.

Generally, the information handling systems used for providing computing and data processing to customers includes a method or facility for recording system usage information. However, in normal operations the usage information is not directly associated with specific jobs or unit of work and their access codes. To obtain the specific cost allocation for internal charge-back to a particular customer, the system usage information has to be correlated to the customer and to the access code associated therewith. Under prior systems, obtaining an accurate and timely cost allocation was particularly difficult as the information handling systems did not allocate the usage information to specific customer and the customer jobs or unit of work were often run on a plurality of information handling systems located at various remote sites. Also, the task of performing complete cost allocation for all customer jobs or units of work was made more difficult due to the large number of customer jobs or unit of work that are performed by service providers and

the unavailability of automated methods for collecting the usage information and allocating it to the specific customer. Traditional methods have also been inflexible and not permit each computing and information handling site the ability to identify unique criteria associated with the job or unit of work as a basis for determining access codes. This flexibility is especially important for any globally operating corporation that has diverse and remotely located operating centers that may have unique data processing requirements and special billing processes.

SUMMARY OF THE INVENTION

Therefore, a need has arisen for a system for deriving job access codes that overcomes the disadvantages and deficiencies of the prior art by meeting the need to assign access codes with the flexibility to specify unique criteria for assigning access codes.

In accordance with the present invention, such a system and a method for doing the same are provided which eliminate or substantially reduce the disadvantages associated with prior systems.

In one aspect of the invention, a system for deriving financial responsibility identification includes a load process receiving a set of system attributes and using at least one of the system attributes for looking up in at least one lookup table for an access code and at least one formula value, and further storing the access code and at least one formula value in an access code lookup table. The system further includes a derivation process which receives a specification of values for the set of system attributes from a billing record, uses the specification of values for looking up in the at least one lookup table for a derivation formula, uses the derivation formula for looking up in the billing record for an attribute value, and further uses the attribute value for comparing with the formula value stored in the access code lookup table. The access code associated with the formula value comparable to the attribute value is the derived access code.

In yet another aspect of the invention, a method for deriving access codes for billing the customer includes the steps of first receiving a set of system attributes and using at least one of the system attributes for looking up in at least one lookup table for an access code and at least one formula value, and then storing the access code and at least one formula value in an access code lookup

table. A subsequent step then receives a specification of values for the set of system attributes from a billing record, which is then used for looking up in the at least one lookup table for a derivation formula. The derivation formula is then used for looking up in the billing record for an attribute value, which is compared with the formula value stored in the access code lookup table. The access code associated with the formula value comparable to the attribute value is the derived access code.

BRIEF DESCRIPTION OF THE DRAWINGS

For a better understanding of the present invention, reference may be made to the accompanying drawings, in which:

FIGURE 1 is a simplified block diagram of an embodiment of the system and method for deriving access codes constructed according to the teachings of the present invention;

FIGURE 2 is a more detailed block diagram of an embodiment of a load process or subsystem with its exemplary inputs;

FIGURE 3 is a simplified top-level flowchart of an embodiment of a derivation process or subsystem;

FIGURE 4 is a simplified flowchart providing details on a lookup process;

FIGURE 5 is a simplified flowchart providing details on a lookup process;

FIGURE 6 is a simplified flowchart providing details on a lookup process;

FIGURE 7 is a simplified flowchart providing details on a lookup process;

FIGURE 8 is a simplified flowchart providing details on a lookup process;

FIGURE 9 is a simplified flowchart providing details on a lookup process; and

FIGURE 10 is a simplified flowchart providing details on a lookup value process.

DETAILED DESCRIPTION OF THE INVENTION

The preferred embodiment(s) of the present invention is (are) illustrated in FIGURES 1-10, like reference numerals being used to refer to like and corresponding parts of the various drawings.

Referring to FIGURE 1, a system and method for deriving job access codes are indicated generally at 10 and constructed according to the teachings of the present invention. System 10 includes two primary processes or subsystems, a load process or subsystem 12 and a derivation process or subsystem 14. A primary function of load process 12 is to consolidate all the data and control parameters for derivation process 14 to derive or determine the access code for each job. Load process 12 and derivation process 14 are two separate processes which are generally independently run. For example, load process 12 may execute on demand and derivation process 14 may execute as part of a large accounting system 15, which is typically run on a scheduled periodic basis.

Load process 12 receives a number of inputs, including control tables 16, user-maintained flat files 18, and control parameters 20. Control parameters 20 is a set of system attributes or information that a user may use to identify which records in user-maintained flat files 18 are to be used in access code load process 12. User-maintained flat files 18 contain derivation search values and associated access codes. Control tables 16 includes information about the records in user-maintained flat files 18, including the type of records, and the offset and length of the fields in each record. Control tables 16, control parameters 20, and user-maintained flat files 18 are generally entered by users via a user interface (not shown), and may be accessed and modified in this manner. Load process 12 also receives data stored in two tables,

main table 22 and formula table 24, in order to populate two other tables, value mask table 26 and value range table 28. Tables 22-28 may be stored and maintained in relational databases, such as DB2. Data from user-maintained flat files 18 are used to populate value range and value mask tables 28 and 26, which are subsequently used in derivation process 14.

Derivation process 14 receives data extracted from interim journal tables 30, main table 22, formula table 24, value mask table 26, and value range table 28 as input to derive the proper access code for each job or unit of work. Journal tables 32 updated with the derived access code are then provided as output. Journal tables 32 updated with the proper access codes are then used for billing and charge-back by accounting system 15.

Referring to FIGURE 2, a more detailed block diagram of load process 12 is shown with exemplary inputs.

Set forth below are column definitions for selected exemplary columns in tables 22-28 containing exemplary inputs and other attributes. It is important to note that these tables may contain additional columns not listed or described herein to further aid in the load and derivation processes 12 and 14.

In Main Table 22:

ADRKEY - contains a system assigned value used for associating a row in main table 22 with a row containing corresponding values in value mask and/or range tables 26 and 28.

DERVTYPE- contains a value used to associate a row in main table 22 to row in formula table 24.

LSYSID - logical system identifier associated with the billing activity.

ORDERNO - a number specifying a priority in which the derivation formula identifier (DERVFORM) is applied.

SOFTWARE- specifies the software that generated the interim journal tables for billing purposes.

SUBSYS - subsystem identifier used to further identify the system consuming the resources.

In Formula Table 24:

ACCFORM - contains the formula used to obtain the access code for loading value mask and range tables 26 and 28.

DERVFORM- contains the formula used to identify the billing record field(s) in interim journal tables 30 for its value then used to compare with the value range (VALUELO and VALUEHI) and/or value mask (MASK) values.

DERVTYPE- contains a value used to associate a row in formula table 24 to row(s) main table 22.

RECNAME - contains the name of the record for which values for a particular DERVTYPE can be loaded by load process 12 in batch mode.

VALUE1 - contains the formula used to obtain the first of two values for use in load process 12, and can be the MASK value for value mask table 26 or the VALUELO value in value range table 28.

VALUE2 - contains the formula used to obtain the second of two values for use in load process 12, which is VALUEHI in value range table 28.

In Value Mask Table 26:

ACCESS - contains the access code linking resource usage to a business unit for billing or charge-back purposes.

ADRKEY - contains a system assigned value used for associating a row in main table 22 with a row containing corresponding values in value mask and range tables 26 and 28.

MASK - contains the value to match with exactly or partially during derivation process 14 to derive the access code.

In Value Range Table 28:

ACCESS - contains the access code linking resource usage to a business unit for billing or charge-back purposes.

ADRKEY - contains a system assigned value used for associating a row in main table 22 with a row containing corresponding values in value mask and range table 28.

VALUELO- the low end of the range into which a value must fall to match during derivation process 14 to derive the access code.

VALUEHI- the high end of the range into which a value must fall to match during derivation process 14 to derive the access code.

Control parameters 20 are received as an input by load process 12 to identify which user-maintained flat file(s) 18 to use to load derivation information. Control parameters 20 may include example system attributes such as LSYSID (logical system ID), SUBSYS (subsystem), and SOFTWARE (the software that generated the billing records). Selected or all of these system attributes may be used to

find one or more row entries in main table 22 to derive a corresponding DERVTYPE. For example, assume that the control parameters produced a match with DERVTYPE = 'DSNAUTH'. The DERVTYPE or 'DSNAUTH' from main table 22 is then used to find a row entry in formula table 24 with the same DERVTYPE. In the example shown in FIGURE 2, the first row of formula table 24 has 'DSNAUTH' as the DERVTYPE. That same row also has 'ADV' as the RECNAME (record name), 'ADVVAL01' as VALUE1, 'ADVVAL02' as VALUE2, and 'ADVACCSS' as ACCFORM (access code formula). The ACCFORM value 'ADVACCSS' in formula table 24 is then used to find a match in control tables record 16 having ADV RECNAMEs. This is shown under the ADV heading as ADVACCSS (6,4), which represents the data length and offset in a predetermined unit, such as bytes, into an appropriate record in user-maintained flat files 18 to obtain the access code for the row entry in main table 22 marked with ADRKEY = 11111 that will be stored in either the value mask table 26 or value range table 28. At offset 4 in user-maintained flat file 18 DSNAUTH record is the access code 'BRRE95'. Formula table 24 also includes VALUE1 and VALUE2, which have the values 'ADVVAL01' and 'ADVVAL02', respectively. When control tables 16 are searched, 'ADVVAL01' yields (60,10) and 'ADVVAL02' yields (60, 70) as the lengths and offsets for two numerical values '1' and '5' stored in flat files 18. Because VALUE2 has a value, the access code 'BRRE95' is entered into value range table 28 in the ACCESS column at row ADRKEY = 11111 (matching main table row entry) because '1' and '5' indicate a range. The values '1' and '5' are also entered into value range table 28 in the same row as VALUELO and VALUEHI, respectively.

Using another example, assume that the control parameters entered by the user yielded the row with ADRKEY = 22222 as a match. The DERVTYPE, 'CICUSID', is then used

to find a match in formula table 24, which yields RECNAME = 'A2X', ACCFORM = 'A2XACCSS', and VALUE1 = 'A2XUSID'. Note that VALUE2 does not have an entry. Looking in the control table records associated with RECNAME 'A2X' yields A2XUSID of data length 8 and at offset 4. Using the offset and data length information, the value 'BSSJJE' is obtained from an A2X record of user-maintained flat files 18. Further, the value 'BRRE95' is also obtained from the same record using data length 6 and offset 12 information from control tables 16. Since only VALUE1 has a value, the formula is for a value mask, directing the selected access code 'BRRE95' and VALUE1 'BSSJJE' to be entered in value mask table 26 in columns ACCESS and MASK, respectively.

Operating in this manner, value range and mask tables 28 and 26 are populated during load process 12 according to the control parameter input. These tables, along with main and formula tables 22 and 24, are used in derivation process 14 to derive the access codes.

Referring to FIGURE 3, which shows a top-level flowchart of an embodiment of derivation process 14 (FIGURE 1), all or selected system attributes for a job or unit of work are extracted from interim journal tables 30 and passed to derivation process 14, as shown in block 100. An iterative lookup process is performed to find an exact match or a partial match of the passed system attributes in row entries of main table 22 to yield an associated access code, as shown in block 102. If a match is not found (as determined in block 104), and if there are no more tables to be searched, (as determined in block 106) then a blank access code is returned and a good return code variable is set to indicate that no valid access code is found. However, if an exact or partial match is found in block 104, then the associated access code is validated, as shown in block 110. If the access code is not valid as

determined in block 112, execution returns to block 106 to search other tables, if any; otherwise if the access code is valid, it is returned along with the good return code, as shown in block 114.

The perform lookup process 102 is an iterative process that attempts to first find an exact match; if unsuccessful, it then proceeds with more generic and less specific values for the system attributes. Details of the lookup process is described in FIGURES 4-10.

Referring first to FIGURE 4, block 200, a lookup is first performed with system attributes LSYSID, SOFTWARE, and SUBSYS, each with specific values as passed from interim journal tables 30 or billing records (FIGURE 1). If one or more records are located in main table 22 that has the exact values as the passed attributes, as determined in block 202, then an access code value lookup is performed in block 204. Multiple records may have the same attribute values but are prioritized by the ORDERNO parameter (FIGURE 2). Details of this process are shown in FIGURE 10 and described below. If an access code is found (block 206), then execution returns to block 110 (FIGURE 3) to validate the derived access code. If no access code is found, then subsequent records with matching attributes are also processed in a similar manner until none are left, as shown in block 208. When lookup using specific values of LSYSID, SOFTWARE, and SUBSYS does not produce a valid access code, less specific search attributes are used, as shown in FIGURES 5-9.

FIGURE 5 shows a search with specific values for LSYSID and SOFTWARE paired with any value of SUBSYS. In other words, a match is produced with any record that has LSYSID and SOFTWARE matching the specified values thereof regardless of what value it has for SUBSYS (as shown in block 210). SUBSYS is allowed to be a wild card or "don't

care". The process is much the same as shown in FIGURE 4. Any found record is used to look up the access code, using priority established by ORDERNO, until all records with a match are processed, as shown in blocks 212-218.

When a valid access code is still not produced, a lookup for records with a specific LSYSID but any value for SOFTWARE and SUBSYS is performed, as shown in block 220 in FIGURE 6. Found records are also processed in a similar manner, as shown in block 222-228.

If still no valid access code is produced, then records with any LSYSID but specific values for SOFTWARE and SUBSYS are searched for, as shown in block 230 of FIGURE 7. Any records found that matched the specific values for SOFTWARE and SUBSYS are then processed in a similar manner as in blocks 232-238.

A further search iteration shown in FIGURE 8 performs lookup for records with a specific value for SOFTWARE and any value for LSYSID and SUBSYS, as shown in block 240. All records found with this search criteria are processed in blocks 242-248.

Finally, as shown in block 250 in FIGURE 9, a lookup using wild cards or don't cares for all three attributes is performed. Found records are processed by looking up the associated access code(s) as shown in blocks 252-258. If no valid access codes are found using the found records, then a bad return code is set and returned, as shown in blocks 260 and 262.

It may be seen from the foregoing that the lookup process is an iterative process that begins with the most specific search criteria to the least specific search criteria. Of those records that matched the search criteria, a priority established by ORDERNO identifies an order in which these records should be looked at to derive the access code.

FIGURE 10 describes the process flow for looking up the access code (lookup value, block 270) using formula table 24, value mask table 26, and value range table 28 (FIGURE 2). Using the associated value for DERVTYPE in the main table record to lookup a specific row in formula table 24, field name(s) specified by DERVFORM (derivation formula) associated with that DERVTYPE is identified, as shown in block 272. Note that main table 22 further include a column ORDERNO, which is used to establish a priority among DERVFORMS residing in rows with the same LSYSID, SOFTWARE, and SUBSYS values, so that the DERVTYPE of row with the lowest ORDERNO is used first to identify the field in the billing records in interim journal tables 30.

Using the specified field name(s) or DERVFORM from formula table 24, value(s) for the specified field(s) in interim journal tables 30 are read, as shown in block 274. DERVFORM may reference constant values or the following exemplary journal table columns:

1. SUBSYS (subsystem identifier)
2. WORKUNIT (work unit)
3. LSYSID (logical system identifier)
4. ACCESS (access code)
5. DSNAME (dataset name)
6. FIELDN (where N may be any interger value)
7. VOLSER (volume serial number)

Subsequently, the field values are compared with MASK in value mask table 26 and/or VALUELO and VALUEHI values in value range table 28. The MASK value in value mask table 26 may contain wild characters so that only a partial match is required. The match with VALUELO and VALUEHI in value range table 28 requires the field value to be within the

range specified by VALUELO and VALUEHI. The access code contained in the row having the matching value range and/or value mask is then returned as the access code to use for the current billing record, as shown in block 278.

The flexibility of the instant system and method is imparted by the manner in which an operating center or site may specify the control parameters, value formula, value ranges and value masks, and derivation formula, all of which play a part in determining what the resultant access code is. An operating center is thus free to tailor its access code assignment to its unique business or data processing operations.

According to an embodiment of the present invention, a method for deriving access codes for billing customers is shown which comprises the steps of receiving a set of system attributes and using at least one of the system attributes for looking up in at least one lookup table for an access code and at least one formula value, storing the access code and at least one formula value in an access code lookup table, receiving a specification of values for the set of system attributes from a billing record, using the specification of values for looking up in the at least one lookup table for a derivation formula, using the derivation formula for looking up in the billing record for an attribute value, using the attribute value for comparing with the formula value stored in the access code lookup table, and generating the access code associated with the formula value comparable to the attribute value. The step of looking up in a control table may further include the step of looking up at least one value mask. Further, the step of looking up in a control table may further include the step of looking up at least one value mask containing a wild character string. The step of looking up in a control

table may further include the step of looking up two values indicative of a value range. The step of receiving the set of system attributes may further include the step of receiving a logical system identifier, a subsystem identifier, and an identification of a software which generated the billing records. Moreover, the method may further comprise the step of producing a billing record with the generated access code inserted therein.

Although the present invention and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations can be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

WHAT IS CLAIMED IS:

1. A computer system for deriving access codes for billing customers, comprising:

a load subsystem receiving a set of system attributes and using at least one of said system attributes for looking up in at least one lookup table for an access code and at least one formula value, and storing said access code and at least one formula value in an access code lookup table; and

a derivation subsystem receiving a specification of values for said set of system attributes from a billing record and using said specification of values for looking up in said at least one lookup table for a derivation formula, using said derivation formula for looking up in said billing record for an attribute value therefrom, using said attribute value for comparing with said formula value stored in said access code lookup table, and determining said access code associated with said formula value comparable to said attribute value.

2. A computer system for deriving access codes for billing customers, comprising:

at least one lookup table;

a load subsystem receiving said set of system attributes and using at least one of said system attributes for looking up in said at least one lookup table for a corresponding lookup key, and using said lookup key for looking up in said at least one lookup table for a corresponding access code formula label and at least one formula value label, further using said access code formula label and at least one formula value label for looking up in said at least one lookup table for a location of corresponding access code and at least one formula value stored in said plurality of user files, and storing said obtained access code and

at least one formula value in said at least one lookup table; and

a derivation subsystem receiving a specification of values for said set of system attributes from a billing record and using said specification of values for looking up in said at least one lookup table for said lookup key, using said lookup key for looking up in said at least one lookup table for a corresponding derivation formula label, using said derivation formula label for looking up in a billing record for an attribute value, using said attribute value for comparing with said formula value stored in said at least one lookup table, and determining said access code associated with said formula value comparable to said obtained value.

3. The computer system, as set forth in Claim 2, wherein said at least one lookup table comprises a control parameters file containing a plurality of values for a set of system attributes specifying information about at least one system or device which performed the customer unit of work.

4. The computer system, as set forth in Claim 2, wherein said at least one lookup table comprises a plurality of user files each containing a plurality of access codes and associated at least one formula value.

5. The computer system, as set forth in Claim 2, wherein said at least one lookup table comprises a control table containing a location configuration of access codes and associated at least one formula value stored in said plurality of user files.

6. The computer system, as set forth in Claims 1 or 2, wherein said at least one lookup table comprises a main lookup table having said set of system attributes as column entries in addition to column entries of a first lookup key.

7. The computer system, as set forth in Claim 6, wherein said at least one lookup table comprises a formula lookup table having said first lookup key as a column entry in addition to column entries of an access code formula label, a derivation formula label, and at least one formula value label.

8. The computer system, as set forth in Claim 7, further comprising:

a control parameters file containing a plurality of values for a set of system attributes specifying information about at least one system or device which performed the customer unit of work;

a plurality of user files each containing a plurality of access codes and associated at least one formula value; and

a control table containing a location configuration of access codes and associated at least one formula value stored in said plurality of user files.

9. The computer system, as set forth in Claim 8, wherein said load subsystem receives said set of system attributes and uses at least one of said system attributes for looking up in said main lookup table for a corresponding lookup key, uses said lookup key for looking up in said formula lookup table for a corresponding access code formula label and at least one formula value labels, further using said access code formula label and at least one formula value label

for looking up in said control table for a location of corresponding access code and at least one formula value stored in said plurality of user files, and stores said obtained access code and at least one formula value in an access code lookup table.

10. The computer system, as set forth in Claim 9, wherein said derivation subsystem receives a specification of values for said set of system attributes from a billing record and uses said specification of values for looking up in said main lookup table for said lookup key, using said lookup key in said formula lookup table for looking up a corresponding derivation formula label, uses said derivation formula label for looking up in said billing record for an attribute value therefrom, using said attribute value for comparing with said formula value stored in said access code lookup table, and determining said access code associated with said formula value comparable to said attribute value.

11. The computer system, as set forth in Claims 1 or 2, wherein said at least one formula value includes a value mask.

12. The computer system, as set forth in Claim 11, wherein said value mask contains a wild character string.

13. The computer system, as set forth in Claims 1 or 2, wherein said at least one formula value includes two values indicative of a value range.

14. The computer system, as set forth in Claims 1 or 2, wherein said set of system attributes includes a logical system identifier, a subsystem identifier, and an identification of a software which generated said billing records.

15. The computer system, as set forth in Claims 1 or 2, wherein said derivation subsystem produces a billing record with said generated access code inserted therein.

16. A method for deriving access codes for billing customers, comprising the steps of:

(a) receiving a set of system attributes and using at least one of said system attributes for looking up in at least one lookup table for an access code and at least one formula value.

(b) storing said access code and at least one formula value in an access code lookup table;

(c) receiving a specification of values for looking up in said at least one lookup table for a derivation formula;

(d) using said specification of values for looking up in said at least one lookup table for a derivation formula;

(e) using said derivation formula for looking up in said billing record for an attribute value;

(f) Using said attribute value for comparing with said formula value stored in said access code lookup table; and

(g) generating said access code associated with said formula value comparable to said attribute value.

17. The method, as set forth in Claim 16, wherein said steps (a) through (b) are performed by a load process, and said steps (c) through (g) are performed by a derivation process, said load and derivation processes being executed independently.

18. The method, as set forth in Claim 16, wherein said step (a) further comprises the steps of:

using at least one of said system attributes for looking up in a main lookup table for a corresponding lookup key;

using said lookup key for looking up in a formula lookup table for a corresponding access code formula label and at least one formula value labels; and

using said access code formula label and at least one formula value label for looking up in a control table for a location of corresponding access code and at least one formula value stored in said plurality of user files.

19. The method, as set forth in Claim 18, wherein said step (d) further comprises the steps of:

using said specification of values for looking up in said main lookup table for said lookup key; and using said lookup key in said formula lookup table for looking up in a corresponding derivation formula label.

20. The method, as set forth in Claim 16, wherein said step of receiving a specification of values for said set of system attributes includes the step of receiving a specification of values for a logical system identifier, a subsystem identifier, and an identification of a software which generated said billing records.

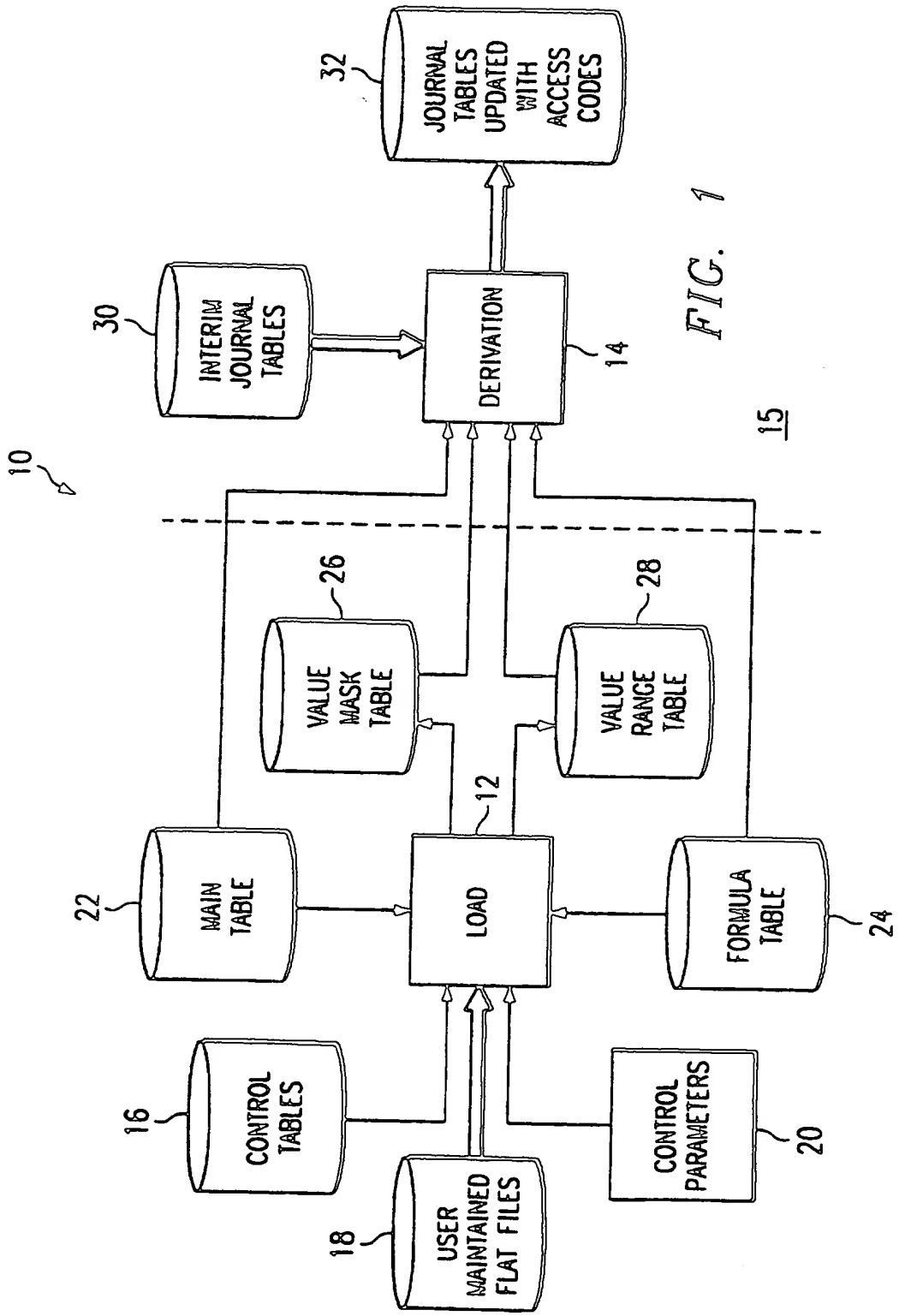
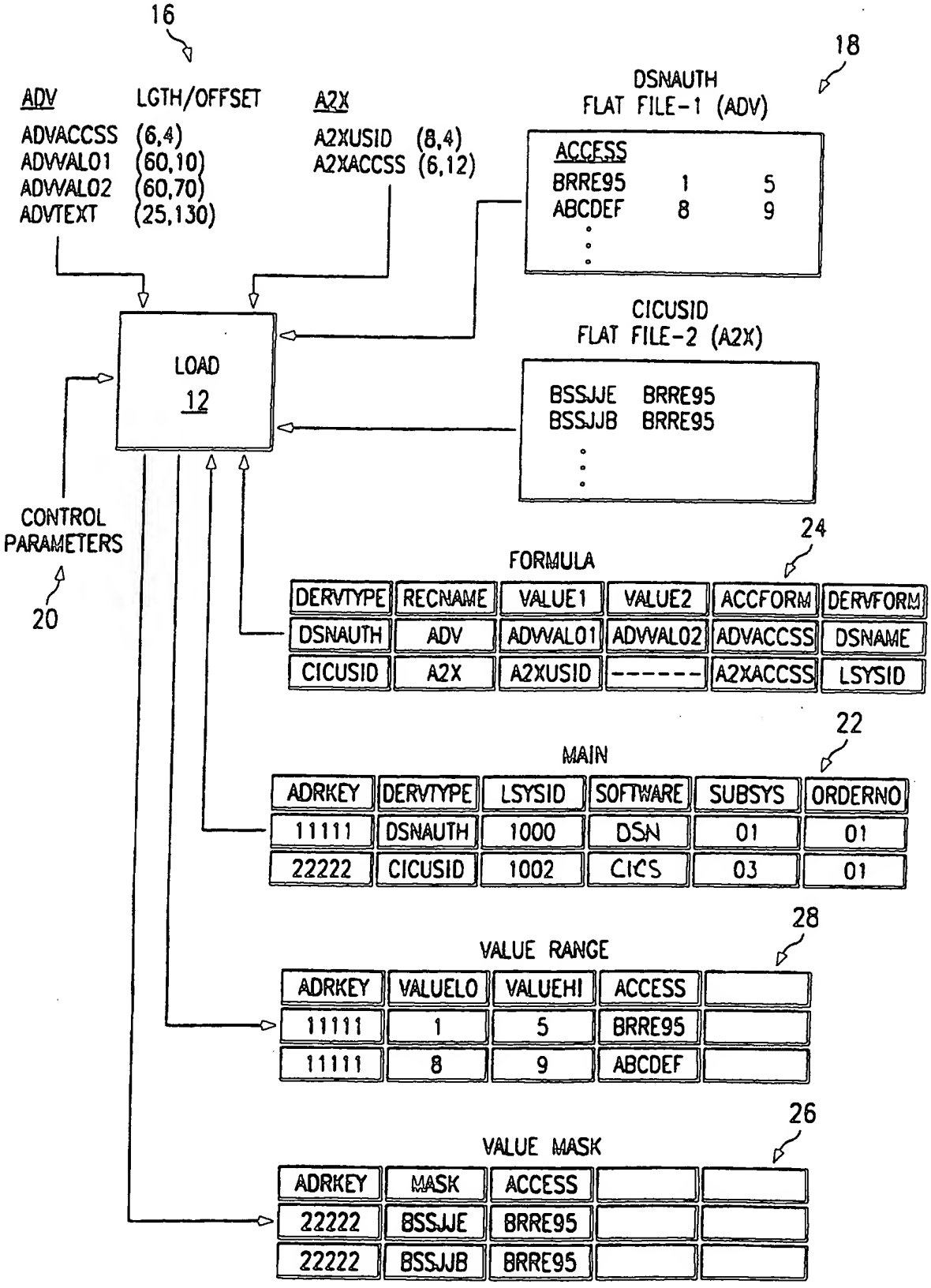

JOURNAL TABLES UPDATED WITH ACCESS CODES
32
INTERIM JOURNAL TABLES
30
DERIVATION
14
10
15
VALUE MASK TABLE
26
VALUE RANGE TABLE
28
12
MAIN TABLE
22
LOAD
FORMULA TABLE
24
CONTROL TABLES
16
CONTROL PARAMETERS
20
USER MAINTAINED FLAT FILES
18

FIG. 1

FIG. 2


16
18
ADV
LGTH/OFFSET
A2X
ADVACCSS (6,4)
ADVVAL01 (60,10)
ADVVAL02 (60,70)
ADVTEXT (25,130)
A2XUSID (8,4)
A2XACCSS (6,12)
DSNAUTH
FLAT FILE-1 (ADV)
ACCESS
BRRE95 1 5
ABCDEF 8 9
LOAD
12
CICUSID
FLAT FILE-2 (A2X)
BSSJJE BRRE95
BSSJJB BRRE95
CONTROL
PARAMETERS
20
24
FORMULA
DERVTYPE | RECNAME | VALUE1 | VALUE2 | ACCFORM | DERVFORM
DSNAUTH | ADV | ADVVAL01 | ADVVAL02 | ADVACCSS | DSNAME
CICUSID | A2X | A2XUSID | ------ | A2XACCSS | LSYSID
22
MAIN
ADRKEY | DERVTYPE | LSYSID | SOFTWARE | SUBSYS | ORDERNO
11111 | DSNAUTH | 1000 | DSN | 01 | 01
22222 | CICUSID | 1002 | CICS | 03 | 01
28
VALUE RANGE
ADRKEY | VALUELO | VALUEHI | ACCESS
11111 | 1 | 5 | BRRE95
11111 | 8 | 9 | ABCDEF
26
VALUE MASK
ADRKEY | MASK | ACCESS
22222 | BSSJJE | BRRE95
22222 | BSSJJB | BRRE95

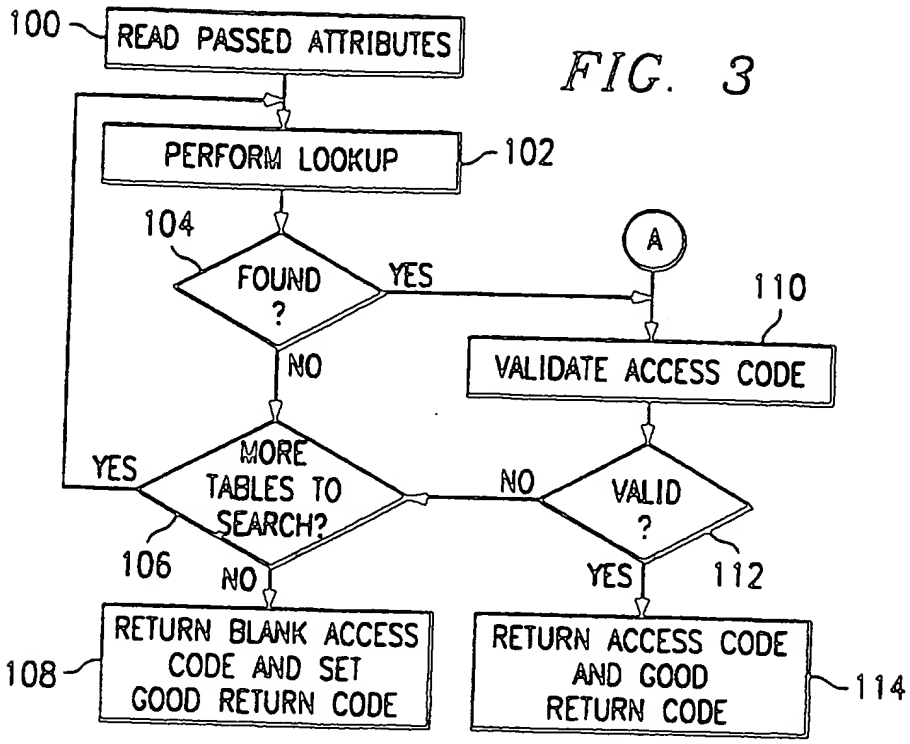

100
READ PASSED ATTRIBUTES
PERFORM LOOKUP
102
104
FOUND ?
YES
A
110
VALIDATE ACCESS CODE
NO
YES
MORE TABLES TO SEARCH?
NO
VALID ?
106
NO
YES
112
108
RETURN BLANK ACCESS CODE AND SET GOOD RETURN CODE
RETURN ACCESS CODE AND GOOD RETURN CODE
114

FIG. 3

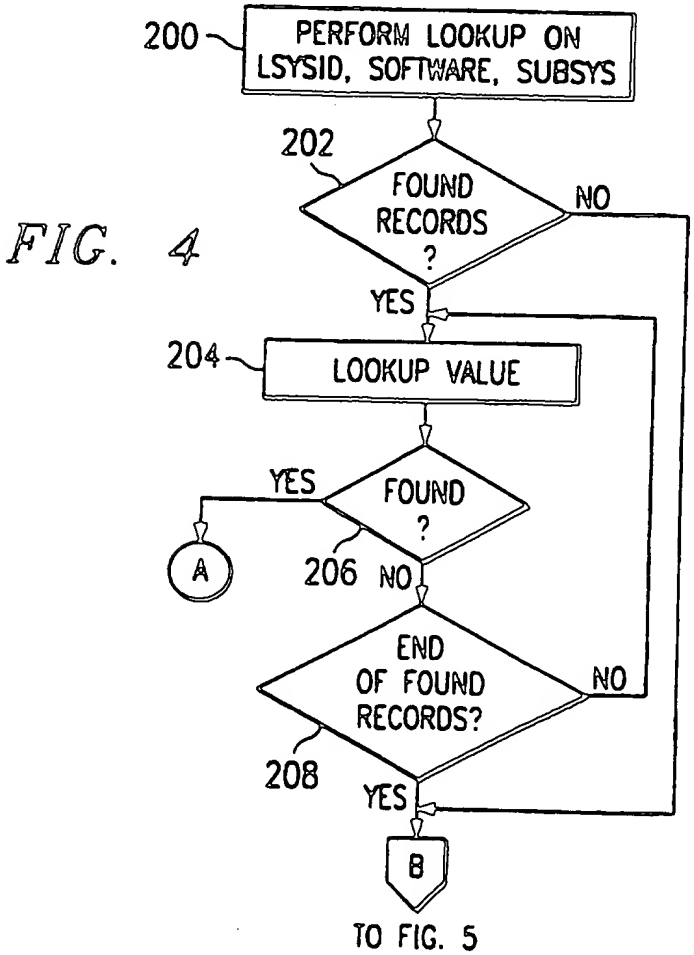

200
PERFORM LOOKUP ON LSYSID, SOFTWARE, SUBSYS
202
FOUND RECORDS ?
NO
YES
204
LOOKUP VALUE
YES
FOUND ?
A
206
NO
END OF FOUND RECORDS?
NO
208
YES
B
TO FIG. 5

FIG. 4

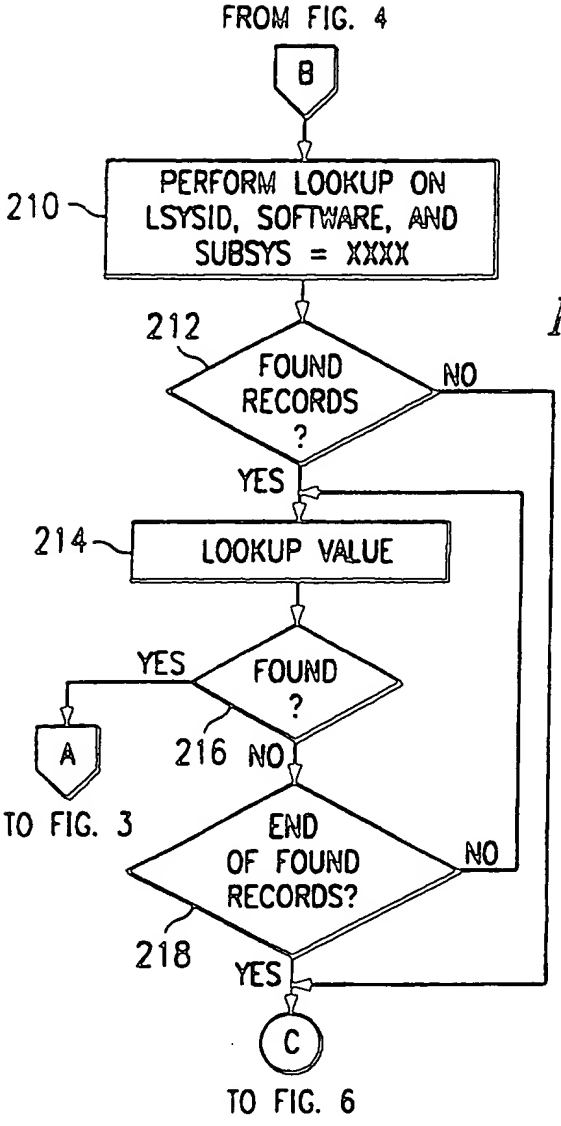

FROM FIG. 4
B
210
PERFORM LOOKUP ON LSYSID, SOFTWARE, AND SUBSYS = XXXX
212
FOUND RECORDS ?
NO
YES
214
LOOKUP VALUE
YES
FOUND ?
A
216
NO
TO FIG. 3
END OF FOUND RECORDS?
NO
218
YES
C
TO FIG. 6

FIG. 5

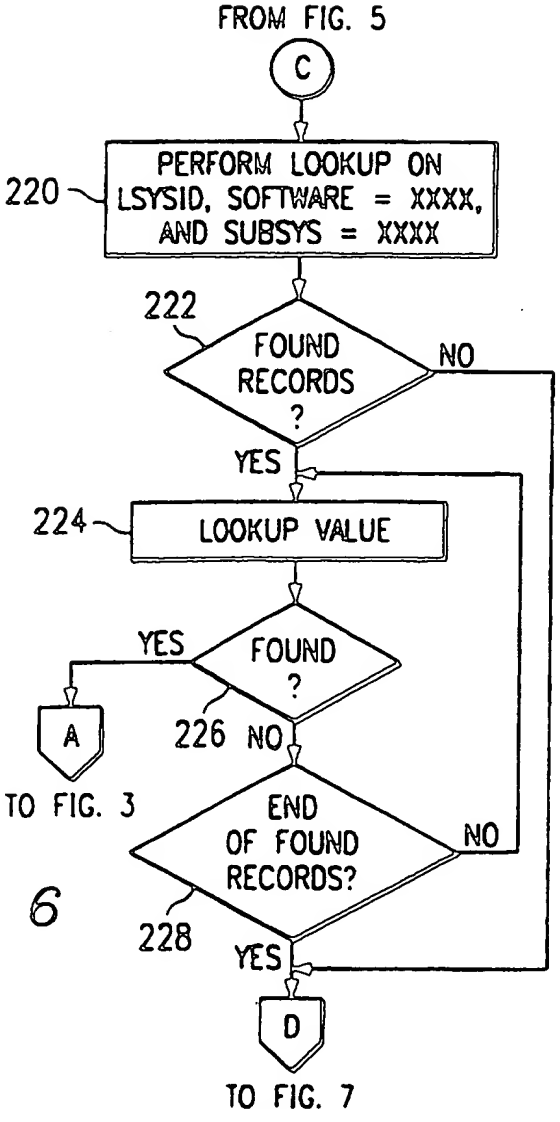

FROM FIG. 5
C
220
PERFORM LOOKUP ON LSYSID, SOFTWARE = XXXX, AND SUBSYS = XXXX
222
FOUND RECORDS ?
NO
YES
224
LOOKUP VALUE
YES
FOUND ?
A
226
NO
TO FIG. 3
END OF FOUND RECORDS?
NO
228
YES
D
TO FIG. 7

FIG. 6

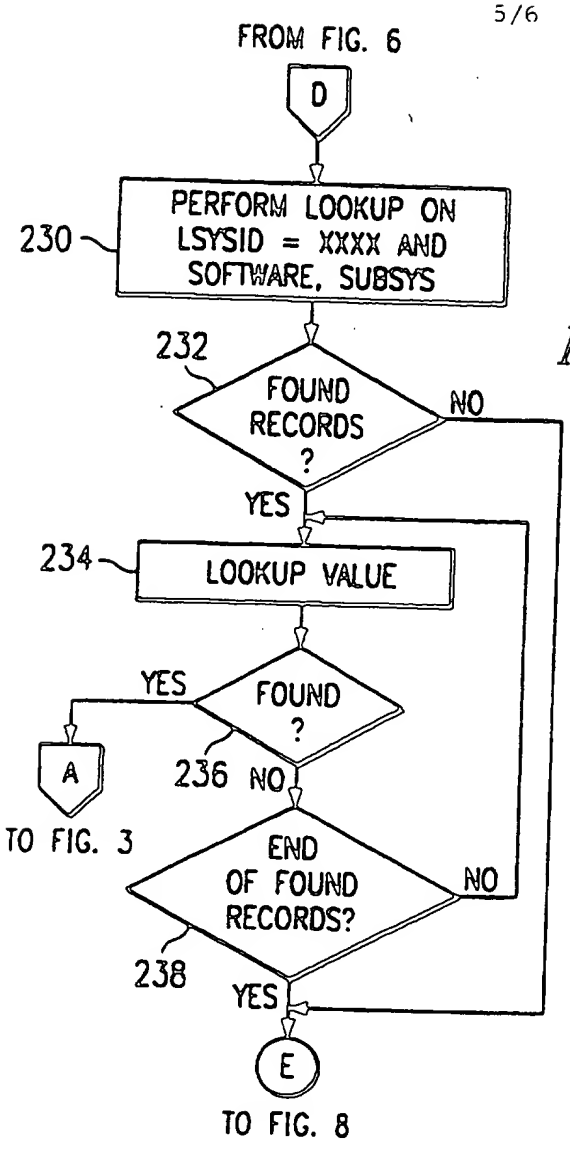

FROM FIG. 6
D
PERFORM LOOKUP ON LSYSID = XXXX AND SOFTWARE, SUBSYS
230
232
FOUND RECORDS ?
NO
YES
234
LOOKUP VALUE
YES
FOUND ?
A
TO FIG. 3
236
NO
END OF FOUND RECORDS?
NO
238
YES
E
TO FIG. 8

FIG. 7

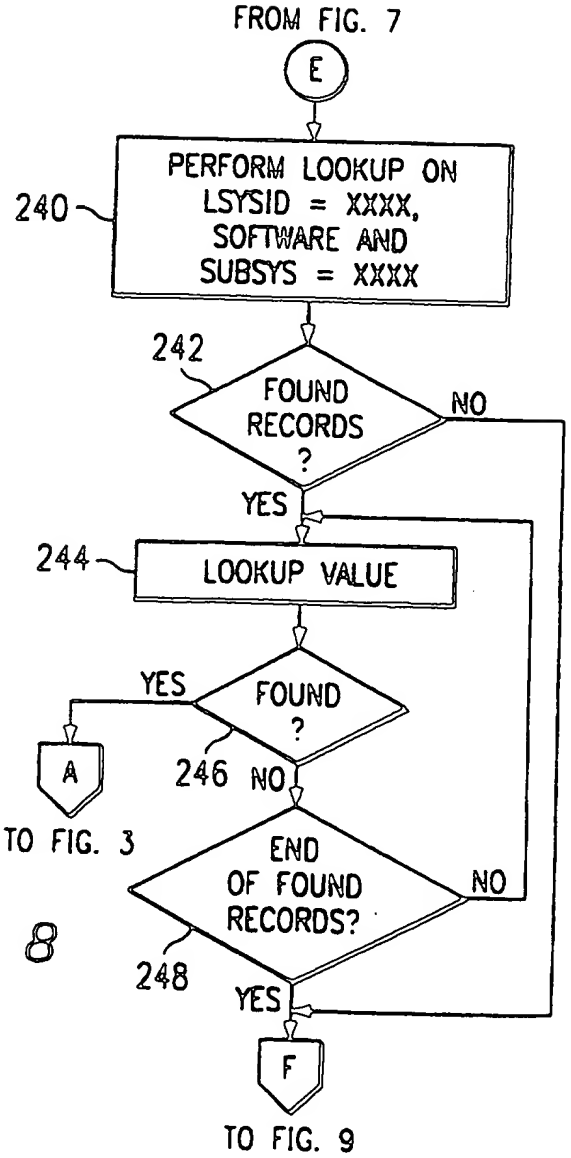

FROM FIG. 7
E
PERFORM LOOKUP ON LSYSID = XXXX, SOFTWARE AND SUBSYS = XXXX
240
242
FOUND RECORDS ?
NO
YES
244
LOOKUP VALUE
YES
FOUND ?
A
TO FIG. 3
246
NO
END OF FOUND RECORDS?
NO
248
YES
F
TO FIG. 9

FIG. 8